

Object Oriented Programming From Daily Life Perspective

Thanh Tam Vo, 103487596
Faculty of Science, Engineering, and Technology
Swinburne University of Technology
Ho Chi Minh, A35 Bach Dang, Tan Binh District

ABSTRACT

Object-Oriented Programming (OOP) is a core unit that every student who is majoring in *Information Technology* (IT) or *Computer Science* (CS) must complete. This concept is applied to the programming world such as game design, neural networking, parallel computing, etc., however, in this paper, I will present the connection between the fundamental Object-Oriented Programming concept such as Abstraction, Encapsulation, Polymorphism, and Inheritance, and the using of mobile communication technology.

1. INTRODUCTION

Almost university in the world has a prerequisite for the unit Object-Oriented Programming since students are required to have a basic understanding of procedural programming by working with variables, arrays, algorithms, etc. However, an algorithmic insight requires a lot of effort, and curiosity that all students cannot adapt immediately, especially for those who just have transitioned from high school to college. Depending on the computing experience, students can achieve a high score for the course *Introduction to Programming (or similar)*, which is the mandatory course that must be completed before enrolling in the *OOP* course in any IT/CS syllabus, or they can still struggle with that unit, but somehow, they still passed the course. This circumstance is alarming because in *OOP*, the theory is more abstract and coding is not the only skill needed, but also the design pattern for organizing every piece of object in the software must be considered.

Learning *OOP* is challenging, especially for some terminology such as *Abstraction*, *Encapsulation*, *Polymorphism*, and *Inheritance* explaining theoretically will make the student confused. In order to help the student achieving a deep understanding of Object-Oriented Programming, in this paper, I will discuss the four fundamental OOP concepts above that we can easily observe in daily life, but at first, a bit theory will be covered so that we can have an overview of OOP concepts before digging into our day-to-day life.

The methodology in this paper is analyzing the construction of mobile communication that we can see every day, and breaking them down into pieces, from here, the connection to OOP concepts will be presented.

Keywords: OOP, Mobile Communication Network, Network Generation

2. BACKGROUND KNOWLEDGE

In this section, the concept of OOP and its four-pillar will be explained theoretically.

2.1 Object-Oriented Programming

According to Alexander S. Gillis, Object-Oriented Programming is a computing approach to organize programming object by its attributes and behaviors, instead of focusing more on computing logic, programmers want to manipulate the data related to the objects, therefore the essential design pattern in OOP is dividing objects into groups, or in programming words: classes.

In Object-Oriented Programming, there are four principles that need to be pointed out: *Encapsulation, Abstraction, Inheritance, and Polymorphism.*

2.2 Class

Each programming object is presented by a class, which is a collection of variables, functions, and properties.

Eg: a class called Human has a variable *name*, *age* and method (function) *breathe*

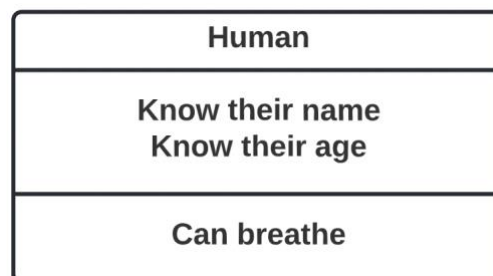


Figure 1. Human class

2.3 Encapsulation

Encapsulation is the process of manipulating data accessibility of the class. In the programming language C#, there are some modifiers to manage the visibility of each attribute of the object.

- **public:** the attribute can be accessed from any code that in the same namespace
- **private:** the attribute can be accessed through the property of the class; the derived class even cannot access this attribute. To solve this situation, we use the *protected* modifier.
- **protected:** this modifier performs as the *private* modifier, however if a base class has a protected attribute, the derived class can access that attribute.
- Etc.

2.4 Abstraction

Abstraction is the term to present necessary information of the class but do not show the way that information being implemented.

Eg: below is an abstract class called EspressoCoffeeMachine which is written in C#:

```
public abstract class EspressoCoffeeMachine {
    public EspressoCoffeeMachine()
    {
    }
    public abstract void MakeCoffee();
}
```

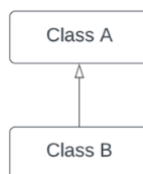
The class above let the user know that it can make a coffee but does not show how it works.

2.5 Inheritance

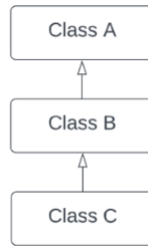
In Object-Oriented Programming, we can create a new class based on an existing class, the new class will inherit all features and characteristic from the base class. As we discussing before, the derived class (child class) cannot access the attribute with *private* modifier.

There are some types of inheritance

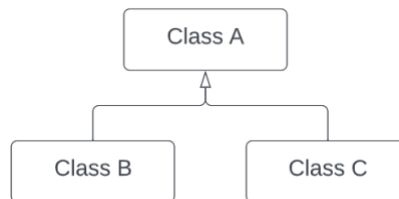
- Single Inheritance



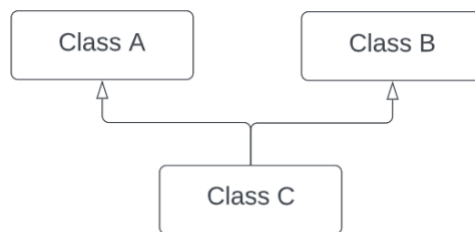
- Multilevel Inheritance



- Hierarchical Inheritance



- Multiple Inheritance



2.6 Polymorphism

Many subclasses can inherit from an abstract base class, so they will have the same method (in the name) but each derived class can perform that method differently. We prefer the word *polymorphism* for this situation.

3. METHODOLOGY

This section will discuss about how a daily phone call being established and show the connection to Object-Oriented Programming concept.

3.1 Mobile communication technology

According to Pew Research Center, the number of calls per day for adults is 5, but have we ever wondered what technology behind this normal action and why do we have many different mobile communication generations such as: 1G, 2G, 3G, and 4G.

3.1.1 Receiving voice and converting into digital signal

The phone will receive your voice as analog signal through the microphone and then convert them into digital signal by Micro Electro-Mechanical System (MEMS) and Integrated Circuits (IC). The Figure 2 below will show the mentioned process.



Figure 2. Signal converting process

3.1.2 Electromagnetic wave

Inside the phone, there is an antenna converting the digital signal of your voice into electromagnetic wave. This electromagnetic wave will be transmitted to the other phone in the call through a cell tower as figure 2.1 below.

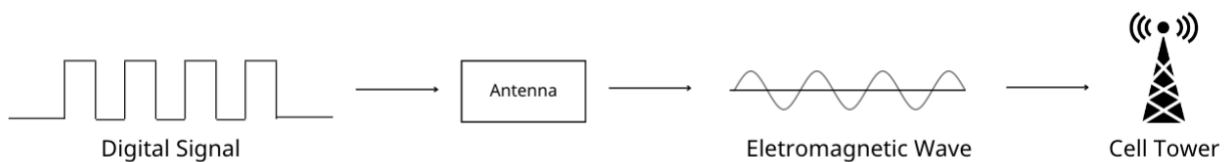


Figure 2.1. Electromagnetic wave transmission

3.1.3 Receiving call

In an established call, if a phone catches the electromagnetic wave from the remain phone (or cell tower), it will receive digital signal, from now, a reverse process will perform turning digital signal into analog signal so that human can understand.

3.1.4 Mobile communication generation

Wireless communication is the process of transferring data or information between two or more than digital devices without using any electrical cable, with the existing of this communicating, our life is getting easier.

A journal named *IJCA-International Journal of Computer Applications* released research of the revolution in Mobile Wireless Technology. Long story shorts, the Figure 2.2 below will show the difference between 4 mobile generations and their special feature.

Mobile Generation	Wireless Transmission	Data Speed	Special Features
First Generation (1G)	Analog Signal	~2.4Kbps	No Security
Second Generation (2G)	Digital Signal	~56Kbps	First SMS messaging
Third Generation (3G)	Digital Signal	~5.8Mbps	Video Calls
Fourth Generation (4G)	Digital Signal	~100Mbps	High Quality Video Streaming

Figure 2.2 Mobile Wireless Network Comparison

3.2 Abstraction concept relating to mobile communication

We all know that with a phone, we can dial a number and call someone, but we do not know how our phone performs these tasks like I demonstrate in the Section 3.1, this is called *Abstraction*, since we just knew what our phone will do (calling, sending the message), but the whole operation system behind performing these tasks, it is not necessary to found out.

Now, let's encode the phone and write it in OOP language as the following Figure 3.

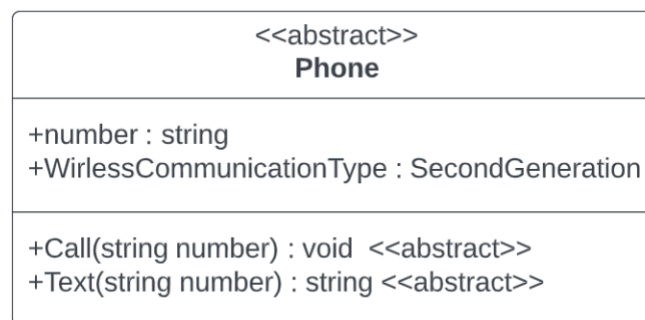


Figure 3. Phone class

3.3 Inheritance and Polymorphism concept relating to mobile communication

As I discussed above, the wireless communication has 4 different generations and they perform the data transfer task with different speed, therefore we can think of inheritance and polymorphism. Now we will create a base class called CommunicationNetwork as Figure 3.1 below

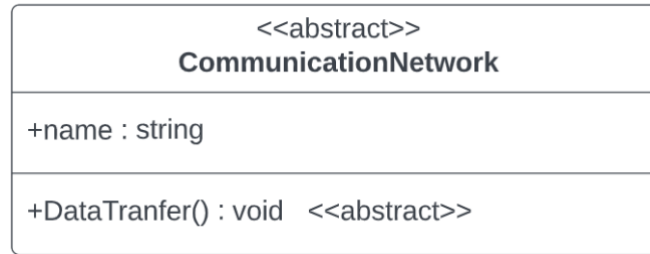


Figure 3.1. CommunicationNetwork class

Next, four generations will be created based on the base class above, the Figure 3.2 will demonstrate this idea

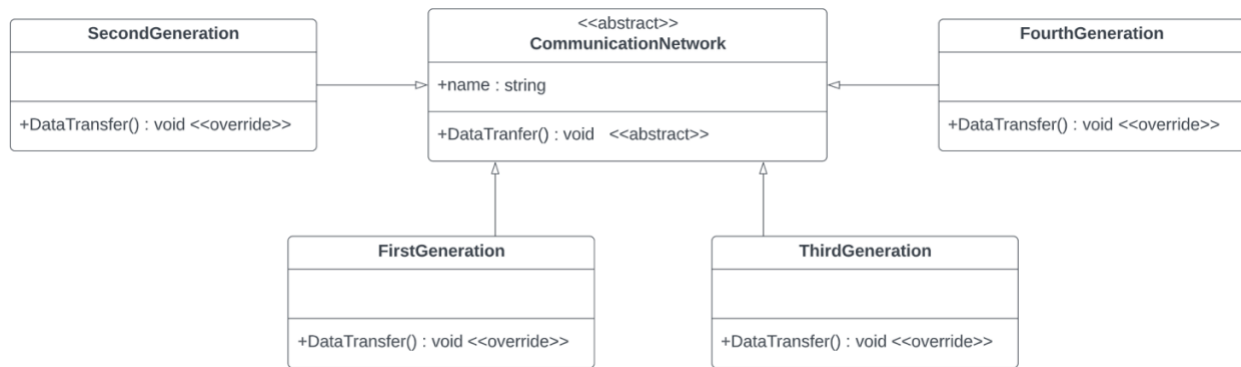


Figure 3.2 Inheritance demonstration

Four derived classes above perform the same task (*DataTranfer()*), however because we all know that their data speed are different, so we can classify them as the using of Polymorphism in Object-Oriented Programming.

3.4 Encapsulation concept relating to mobile communication

In order to protect our phone from data breach, we need a password for our phone. In other words, a class called Phone has a private attribute named *password*, it can not be accessed by other classes. In the mobile communication scenario, if a phone has a call history, only that phone can access its call history. So, we can present an UML diagram like Figure 3.3

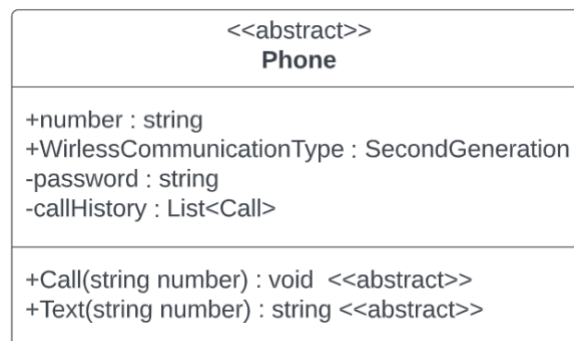


Figure 3.3 Phone class

4. DISCUSSION

Using a mobile communication network explaining the Object-Oriented Programming concept is easier than describing by programming words, however, students need to understand the way the network works so we have Section 3.1 and for who is the freshman in OOP, reading this paper in order to achieve the fundamental knowledge is challenging, since it could not cover an insight of Object-Oriented Programming. There are many essential fields in OOP that this paper does not include such as interface, adapter, Object-Oriented design pattern, etc.

5. CONCLUSION

Talking about Object-Oriented Programming, people usually think of computation without any connection to daily life. However, this paper proves that from a daily life perspective and a little understanding about communication network, an Object-Oriented thinking can be briefly explained. Even for a field such as metropolitan traffic system, we can depict the system as the UML diagram to turn it into OOP concept, however with the limitation in my knowledge, the other future works will explore the metropolitan traffic system topic with the Object-Oriented Programming viewpoint.

REFERENCE

- Bhalla, M.R., & Bhalla, A.V.(2010). Generation of mobile wireless technology survey. International Journal of Computer Application, 5(4), 26-32.
- Gillis, A. S., & Lewis, S. (2021, July 13). *Object-Oriented Programming (OOP)*. SearchAppArchitecture.
<https://www.techtarget.com/searchapparchitecture/definition/object-oriented-programming-OOP>
- Lenhart, A. (2020, August 27). *Cell phones and American adults*. Pew Research Center: Internet, Science & Tech. Retrieved July 30, 2022, from
<https://www.pewresearch.org/internet/2010/09/02/cell-phones-and-american-adults/#:~:text=The%20average%20adult%20cell%20phone,of%20men%20say%20the%20same.>